COPYRIGHT

DISCLAIMER

HARDWARE REQUIREMENTS

To run RAM-DISK you will need the following

1) A Commodore 64 or 64C computer

2) A Commodore 1541 or 1571 disk drive

PRODUCT DESCRIPTION

The RAM-DISK package comprises the following:

1) A 5.25 inch floppy disk in 1541 format containing the software.

2) The RAM-DISK manual (this manual)

R A M - D I S K

C O N T E N T S

Introduction

One of interesting features of the Commodore 64 computer is the large amount of RAM which is hidden behind the ROMs and therefore unavailable to the BASIC user. Unfortunately, most of the time, that RAM lies idle.

In several modern micros, notably the Amiga, Atari ST and the Amstrad a RAM-DISK is supplied as part of the operating system so that users can make effective use of all the memory available. Now, without any additional hardware, this is possible on the 64

In essence RAM-DISK behaves exactly like a **much faster** 1541 or 1571 floppy disk; the only difference being that all the data on the RAM-DISK is held in RAM and not on a physical disk. All the usual disk commands apply, you can look at the directory, LOAD a program, OPEN a file and so on.

As on the Amiga, the Copy command can be used to transfer programs and data files from floppy disk onto the RAM-DISK and, at the end of a session you can dump all the files from RAM-DISK back onto floppy with a single command.

By default, RAM-DISK uses only the hidden memory and takes none away from BASIC. Facilities are provided however to increase or decrease the amount of RAM used and therefore the number of blocks free on the RAM-DISK.

Getting started

Mount the RAM-DISK System disk and LOAD"*",8

You will now be asked DEFAULT LOAD?

to which the answer should normally be Y.

Using RAM-DISK With Other Software

There are a wide range of extension packages available for the C64. Since there is only a limited amount of memory available for use by these packages, the chances of two packages clashing with one another are high. We have done everything possible in designing RAM-DISK to avoid such clashes.

If you try to use RAM-DISK with another package and you run into problems read on.

When you answer Y to the DEFAULT LOAD? question, the software is loaded into the machine in one of a number of possible configurations. If this configuration gives rise to problems you should start over and this time answer N. This will take you into the LOAD CONFIGURATION MENU.

The LOAD CONFIGURATION MENU offers a selection of addresses at which to install the software you have selected. There is no need for you to understand the technical details (although these are displayed for the benefit of the advanced user). The best way to find the load option

(if any) which will allow you to use another package with RAM-DISK is to try them all in turn until you find one which works.
When RAM-DISK is installed it will sign on with the message

```
R "RAM-DISK       " RD 2X
XX Blocks free.
ready.
```

From now on, until you reset the computer, RAM-DISK will respond to all the usual disk commands.

Device numbers

As explained in the C64 System Guide, every device such as the disk drive or printer has a device number. By default, the device number for the floppy disk drive is 8 and all the disk commands make this assumption. If you were to add a second disk drive to the system, you would have to give it a different device number say 9. To look at the directory of the second disk drive you would type LOAD"$",9 and to load a program from it you would type LOAD"prog-name",9 etc. This is exactly how you use RAM-DISK.

Type LOAD"$",9. You should get the (so far empty) RAM-DISK directory. Now type in (or load from floppy disk) a short BASIC program and type SAVE"prog-name",9. If you look at the RAM-DISK directory now, you should see an entry for that file. Type NEW to clear memory and then LOAD"prog-name",9 to get the program back.

NOTE

When you LOAD the directory from a disk drive the directory is loaded into memory as a program, destroying any program currently LOADed. When you LOAD the RAM-DISK directory this does not happen. The directory is displayed but NOT loaded and any program already in memory will be undisturbed.

You will probably find in the course of using RAM-DISK that you frequently need to copy files from floppy disk to RAM-DISK and vica versa. A specially extended Copy command is provided for this purpose, details of which may be found in the command summary at the end of this manual.

Error messages

Error messages are issued by RAM-DISK in exactly the same way as they are by a floppy disk, that is by preparing a message for reading when you INPUT from the command channel. RAM-DISK can however be put into a special mode which causes error messages to be displayed when they occur. For more information on this and the messages generated by RAM-DISK turn to the **Summary of Error Messages** at the end of this manual.

FOR RAM DISK CHANGE 8 TO 9. Ø = ZERO

Extended RAM-DISK commands

Commands are issued to Commodore disk drives by sending a string of text to the drive using a special secondary address of 15, that is by executing a statement like

```
OPEN 1,8,15,"command":CLOSE 1
```

The following command for example scratches the file "temp"

```
OPEN 1,8,15,"S0:temp":CLOSE 1
```

Commands may be sent to RAM-DISK in exactly the same way. They have the form

```
OPEN 1,9,15,"ram-disk-command":CLOSE 1
```

where the second parameter in the open statement is the current device number of RAM-DISK, set by default to 9.

Changing the RAM-DISK device number

This is simple. Type OPEN 1,9,15,"U8":CLOSE1. The RAM-DISK device number is now 8 and the 64 will in future send to RAM-DISK all commands directed to device 8. This is conveniant because it allows any program which uses the disk to read and write files, to access RAM-DISK instead. However, if you wish to access the floppy disk you will have to change the RAM-DISK device number to something different, say device 9 as before, with the command OPEN 1,8,15,"U9":CLOSE 1. Notice that the second parameter of the open statement is now 8, reflecting the (current) device number of RAM-DISK.

IMPORTANT -- During the remainder of this manual, we shall assume that the device number of RAM-DISK has been changed to 8. If you decide not to change the device number to 8 (or you decide to change it to something else) you should of course use the device number you choose in all commands directed to RAM-DISK.

RAM-DISK command formats

For the remainder of this manual (as in the above), all command examples will be given in upper case letters. The purpose of this is to make the text clearer. All commands should be typed in un-shifted characters (without spaces) which will appear on the screen in upper or lower case depending upon which character set is at that moment in force.

Allocating memory to RAM-DISK

When you load RAM-DISK you can select from a range of configuration options. For most purposes the default configuration is suitable. However if you are using certain other software such as a BASIC 4 extension package, you may need to select one of the other options. Whichever option you select, you can at any time instruct RAM-DISK to use additional memory or to use less, by running the supplied program RD-CONFIG on the RAM-DISK system disk.

Blocks and Pages

The C64 contains 64K of RAM in 256 pages each containing 256 bytes. Page 0 starts at location 0, page 1 at location 256 and so on right up to page 255 at location 65280. The RAM between page 8 and page 160 is available to BASIC. If your BASIC program does not use all the memory available, you may wish to increase the number of blocks free on RAM-DISK by reallocating this store. The simplest way to do this is to LOAD"RD-CONFIG" from the system disk. The program will tell you which is the highest page currently allocated to BASIC and ask you to select a new page number. For example, if the top of BASIC is at page 160 (which it is by default), lowering it to page 130 will give you an aditional 30 blocks free on RAM-DISK but will reduce the number of bytes free to BASIC by 30*256=7680. RD-CONFIG is written in BASIC so that you can incorporate the code into your own programs should you wish to do so.

The ALLOCATION commands

In complex applications you may wish to excercise precise control over the pages available to RAM-DISK. By using the following code you can tell RAM-DISK to allocate or deallocate any page in memory. Of course if you allocate a page that is in use by BASIC or the operating system, you should be prepared to crash the computer! The pages used by RAM-DISK do not have to be in a contigious block, any combination of pages is allowed.

To allocate a page, execute the statement

```
OPEN 1,8,15,">"+CHR$(P):CLOSE 1
```

where p is the page number.
To deallocate a page, type

```
OPEN 1,8,15,"<"+CHR$(P):CLOSE 1
```

These commands (or their machine code equivalents) can also be issued from within a program.

Communicating with RAM-DISK from machine code

RAM-DISK works by re-directing the OPEN, CLOSE, LOAD, SAVE, GET A BYTE and PUT A BYTE vectors in high memory.
All RAM-DISK commands can be issued from machine language in exactly the same way as commands are issued to a physical disk drive, details of which are to be found in Commodore's own documentation. To send a command to RAM-DISK (or floppy disk) or to open a file proceed as follows:

1) Set the system variables for the current logical address ($B8), the device number ($BA), the secondary address ($B9), filename length ($B7) filename address ($BB/BC), and the filename string itself to appropriate values.

2) Call OPEN ($FFC0) and then CLOSE ($FFC3).

The procedure is the same for opening a file as it is for sending a command (e.g. Copy) the only difference being that to send a command, the secondary address should be set to 15 and the filename string to the text for the command.

Getting and putting bytes

This is done in exactly the same way as bytes are got or put to any other device, that is by sending a talk or listen, writing or reading the data and sending an abort i/o.

Cold start and Warm start

There are two initalization entry points provided by RAM-DISK which re-start the system in different ways. These are as follows

1) Start RAM-DISK from cold without allocating any pages
2) Warm start RAM-DISK.

There are circumstances, such as after a hardware reset, when RAM-DISK will lose its links into the operating system. On these occasions you can use the warm start entry point without losing data. This procedure re-establishes RAM-DISK's links with the operating system and closes any open files. Provided the areas used by RAM-DISK have not been corrupted no files will have been lost.

To cold start RAM-DISK type SYS P*256+103
To warm start RAM-DISK type SYS P*256+109

where P is the number of the common RAM page, set by default to 199.

Command summary

The following standard disk commands have the same effect on RAM-DISK as they do on the 1541 or 1571 floppy disk drive.

C (copy), S (scratch) V (validate) N (new) R (rename) I (initalize)

As usual they are issued by sending a string to the command channel with the statement.

```
OPEN 1,8,15,"command":CLOSE 1
```

For a detailed description of the commands, refer to one of Commodore's disk drive manuals.

Additionally, the following commands can be addressed to RAM-DISK and have the same effect on it as they would on a 1541/1571.

OPEN CLOSE LOAD SAVE

pattern matching with ? and * is supported.

RAM-DISK will also respond to a range of extended commands. These must be issued as above by opening a file to the error channel and writing the command to it. The following is a list of the allowable commands

CMD TYPE	COMMAND	MEANING
UNIT	Un	Change RAM-DISK device number to n
QUIT	q	Quit RAM-DISK
DEBUG	d	Put RAM-DISK into debugging mode
QUIT DEBUG	x	Restore RAM-DISK to normal operation
ALLOCATE A PAGE	>+CHR$(P)	Cause RAM-DISK to make use of page P.
DEALLOCATE A PAGE	<+CHR$(P)	Cause RAM-DISK to stop uing page P.

The extended COPY command

The C (copy) command has been extended to allow copying between RAM-DISK and floppy disk. Any string which begins with an ampersand (&) is taken (without the &) as being the name of a floppy disk file on physical device 8 as follows:

COMMAND	MEANING
C0:file2=0:file1	Copy file1 on RAM-DISK to file2 on RAM-DISK
C0:file2,type=0:&file1	Copy file1 on floppy disk to file2 on RAM-DISK (see note)
C0:&file2=0:file1	Copy file1 on RAM-DISK to file2 on floppy disk
C0:&=0:*	Copy all files on RAM-DISK to floppy disk

NOTE: The command for copying a floppy disk file to RAM-DISK requires the filetype to be appended to the disk filename as shown above. Possible filetypes are **p** (prg), **s** (seq) and **u** (usr). For example, to copy the sequential file "DATA" to RAM-DISK we would use the command

```
OPEN 1,8,15,"CO:DATA=0:&DATA":CLOSE 1
```

In fact it is not necessary to append a filetype when copying programs as prg is the default filetype.

Summary of error messages

Error messages are generated by RAM-DISK in exactly the same way as they are by a floppy disk drive. To read them you can use a short program such as the following just as you can with a physical disk.

```
10 OPEN 1,8,15:INPUT#1,EN,ER$:CLOSE1:PRINT EN;ER$
```

RAM-DISK can however be instructed to print all error messages automatically.

Debugging mode

A possible source of problems is that RAM-DISK does not have an error indicator as does the 1541/1571. It is therefore not possible to see when things are going wrong by looking at the disk drive. Of course programs should always check the error channel, after disk operations, however, to aid program development, RAM-DISK can be put into debugging mode by sending the character 'd' to the error channel. When RAM-DISK is in debugging mode, any error condition will be reported and any running program will be suspended until a key is hit. Debugging mode may be switched off by sending an x to the error channel.

The following is a complete list of RAM-DISK error messages. Where appropriate the error numbers are the same as those elicited by the 1541/1571. RAM-DISK specific error messages have numbers in the range 80 to 85.

MESSAGE	MEANING
00, OK.	All is well
23,FILE CORRUPT.	A page allocated to RAM-DISK has been corrupted.
30,SYNTAX ERROR.	A command is syntactically erroneous.
61,FILE NOT OPEN.	An attempt has been made to access an un-opened file.
62,FILE NOT FOUND.	RAM-DISK can find no trace of the file requested.
63,FILE EXISTS.	An attempt has been made to write a file which has the same name as one already stored on RAM-DISK.
64,FILETYPE MISMATCH.	The file type specified or implied by a command does not match the file of that name on RAM-DISK.
72,NO PAGES.	A command requests more pages than are available.
72,DIRECTORY FULL.	The directory can acommodate no further files.
83,FILE OPEN.	An attempt has been made to open an already open file.
84,READ FILE ONLY.	An attempt has been made to write to a read file.
85,WRITE FILE ONLY.	an attempt has been made to read from a write file.

There are occasions when it is useful to be able to tell the source of an error message, that is, whether it came from RAM-DISK or from floppy disk. RAM-DISK error messages can easily be identified by the period (full stop) which always terminates the text of the message.

Error messages are annunciated by RAM-DISK, just as they are by Commodore floppy disks, with a trailing ,XX,XX. Floppy disk drives use these positions to indicate any track and sector address associated with the error. Under RAM-DISK these positions are always 00,00.

The desktop utilities

When you first load RAM-DISK from tape you are asked whether you wish to have the clock and calculator installed on RAM-DISK. If you answer yes, these utilities will be available as ordinary program files and may be LOADed in the normal way.

The Calculator

The keys on the calculator display are operated as follows:

Calculator on	F1
Calculator off	F2
Calculator quit	F3
CLR	SHIFT CLR/HOME
CE	C
+/-	F4
Help	H
0..9	0..9
+,-,*,/	+,-,*,/

The Clock

Once LOADed and RUN, the clock installs itself at the top of the screen. The clock display appears as follows

```
00:00:00
```

The small square to the left of the time display is a "button" which you "press" to activate the various clock functions.

Setting the time: Move the cursor to the button and hit CONTROL. A pull down menu will appear. Move the cursor to Set Time and type in the time in HHMMSS format.

Setting the alarm: Proceed as if you were setting the time except that when the menu appears, move to the Set Alarm option.

Moving the clock: Move the cursor to the button, and hit the Commodore key. Any movements you now make with the cursor will also move the clock. To leave the clock in the new position hit the Commodore key again.